

‘Relays’ Fan Coil

**Control Module for Fan Coil Units with
On/Off Valves & Relay-Controlled Fan**

User Manual Version: [0.1]_a

www.zennio.com

CONTENTS

Contents	2
1 Introduction	3
2 Configuration.....	4
2.1 General	4
2.2 Valve-Oriented Control	7
2.2.1 Fan Settings	7
2.2.2 Valve Settings	13
2.3 Fan-Oriented Control	15
2.3.1 Fan Settings	15
2.3.2 Valve Settings	19
2.4 Common Settings (Fan)	20
2.5 Common Settings (Valve)	26
2.6 Cyclical monitoring	29
2.7 Scenes.....	31
2.8 Initialisation	34

1 INTRODUCTION

A variety of Zennio devices incorporate the **'Relays' Fan Coil control module**, which allows controlling integrated or external fan coil units where both the opening of the valves and the fan speed can be controlled through **binary outputs** (relays).

Every two pipes make up a circuit where the water flow is controlled by means of **an on/off valve**, whose state can be managed through a **binary output**. This module provides up to two binary outputs, thus making it possible to control a two-pipe fan coil (one valve) or a four-pipe fan coil (two valves).

Typically, the two water circuits of a four-pipe fan coil unit correspond to the **cooling** and **heating** functions (and to the cooling and heating valves, respectively), being therefore both modes available during the device operation. The single water circuit of a two-pipe fan coil unit, on the other hand, may be configured:

- Cooling
- Heating
- Cooling and Heating

No. of pipes	Output	Action
4	Valve Output 1	Cooling Valve
	Valve Output 2	Heating Valve
2	Valve Output 1	Heating Valve
		Cooling Valve
		Heating + Cooling Valve

Table 1 Actions performed by the binary outputs associated to the valve control.

Regarding the fan speed control, up to three binary outputs will be available, being possible to configure them as **commuting relays** (one specific relay for each fan speed) or as **accumulating relays** (the more relays closed, the higher the fan speed).

Please refer to the specific user manual and datasheet of each Zennio device in order to confirm whether this feature is available or not, and for instructions on the device connection and installation.

2 CONFIGURATION

2.1 GENERAL

The first thing that needs to be parameterised is the type of the fan coil unit that the 'Relays' Fan Coil module will be controlling:

- **A four-pipe, two-valve fan coil unit**, which requires the simultaneous management of two independent circuits (one for heating and one for cooling),
- **A two-pipe, one-valve fan coil unit**, which requires the management of a sole circuit (for heating, for cooling, or for both).

Next, it is necessary to configure the **control type**, depending on the element (valve or fan) the main control is addressed to. Selecting one option or the other will significantly alter how the device operates:

- **Control applied to the valves**: the fan coil module main control will be exercised over the valve, thus making the fan state remarkably depend on the actions performed by the valve. The configuration of this control type is detailed in section 2.20.
- **Control applied to the fan**: the fan coil module main control will be exercised over the fan, thus making the valve state remarkably depend on the actions performed by the fan. The configuration of this control type is detailed in section 2.3.

Specific settings are **common to both control types**. These common settings are explained in sections 2.4 and 2.5.

Other than the fan coil type and the control type, the general configuration also involves the following options:

- **Number of Fan Speeds**: up to three fan speeds can be controlled, although it is possible to enable only one or only two, depending on the fan model. This determines the number of relays required for the fan control.
- **Cyclical Monitoring of the Control Values**: see section 2.6.

- **Scenes:** see section 2.7.
- **Fan Coil Always On:** a communication object can be provided for the switch-on and the switch-off of the fan coil control module, as well as its corresponding status object. Alternatively, it is possible to configure the module to maintain the fan coil control permanently switched on.
- **Initialisation:** sets whether, at the start-up of the device, the 'Relays' Fan Coil module should recover the previous state, or on the contrary perform a custom initialisation (see section 2.8).
- **Action at Bus Voltage Failure:** allows configuring whether the module should remain as is when a KNX bus power failure takes places, or alternatively close the valve and turn off the fan.

ETS PARAMETERISATION

After enabling the fan coil module, the ETS tab tree will contain a category named “Fan coil n”, where “n” will correspond with a certain number. Under this category, a tab named “**Configuration**” will always be available, containing the following general parameters.

The screenshot shows the ETS Configuration window for a Fan Coil module. The left sidebar displays a tree structure with 'GENERAL' and 'FAN COIL' expanded, and 'CONFIGURATION' selected. The main area contains the following configuration options:

- Type of Fan Coil: 4 pipes
- Type of Control: Applied to the Valve
- Number of Fan Speeds: 3
- Cyclical Monitoring of the Control Values (e.g. Thermostat): ☐
- Scenes: ☐
- Fan Coil Always On?: ☐
- Initialization: Default
- Action at Bus Voltage Fault: Nothing

Note: It is recommended to set the parameters of this page first

Figure 1. Fan Coil – General Configuration.

- **Fan Coil Type:** “2 pipes” or “4 pipes”.
- **Mode** (only for two pipes): “Heating”, “Cooling” or “Cooling + Heating”.

In case of selecting "Cooling + Heating" or of having configured the fan coil unit as "4 pipes", a one-byte object ("**[FCn] Mode**") will be provided to allow the selection of the desired mode (0 = Cooling; 1 = Heating), as well as the corresponding status object.

- **Control Type:** "Applied to the valve" (see section 2.2) or "Applied to the fan" (see section 2.3).
- **Number of Fan Speeds:** "1", "2" o "3".
- **Cyclical Monitoring of the Control Values:** enables or disables the "Cyclical Monitoring" tab (see section 2.6).
- **Scenes:** enables or disables the "Scenes" tab (see section 2.7).
- **Fan coil Always On:** enables or disables the continuous operation of the fan coil control module. If disabled, a binary object ("**[FCn] On/Off**") will be available for switching the control on and off, as well as the corresponding status object.
- **Initialisation:** "Default" or "Custom". The latter adds a new entry to the tab tree (see section 2.8).
- **Action on Bus Voltage Failure:** "Nothing" or "Close Valve and Turn Off Fan".

Unless the fan coil module has been configured to be always on, an object will be available to turn it on (value "1") or off (value "0"), "**[FCn] On / Off**", as well as its corresponding status object.

2.2 VALVE-ORIENTED CONTROL

In this case, the control will focus on responding to the orders over the valves that may arrive from the bus.

Certain settings must be performed with independence of the control type selected. It is therefore advisable to read sections 2.4 and 2.5 as well.

2.2.1 FAN SETTINGS

The fan control may be **manual** (the fan speed will be controlled externally), **automatic** (the fan speed will be controlled by the module itself), or **both**. The automatic control mode can be configured to depend on the value of certain objects:

- The Continuous **PI Control object**,
- The **difference** between the ambient temperature and the setpoint.

When both, the manual and the automatic control modes are permitted, a communication object will let switching from one control mode to the other, while the corresponding status object will reflect the current mode. It is possible to set in parameters which of the two control modes must be active **after download**, and also a **time-out counter** so the automatic control mode is automatically triggered back after some time with no activity under the manual control mode. This inactivity is referred to the arrival of manual control orders for the fan.

On the other hand, the integrator may configure the **minimum fan speed that is allowed while the valve remains open**. This ensures that the fan remains in motion whenever the valve stays open.

ETS PARAMETERISATION

When the control type has been configured as applied to the valve, the “Fan” tab shows the following specific options (for the remaining parameters, please refer to section ¡Error! No se encuentra el origen de la referencia.).

- **Fan Control Mode.** “Automatic” (section 2.2.1.1), “Manual” (section 2.2.1.2) or “Automatic + Manual” (section 2.2.1.3)

- **Minimum Fan Speed When the Valve is Open:** the available options may depend on the number of speeds configured under “General” (section 2.1). If set to “0”, the minimum speed will not be restricted.

The screenshot displays the Zennio configuration interface for Fan Coil control. The left sidebar shows a tree structure with 'GENERAL', 'FAN COIL', 'CONFIGURATION', 'Fan', and 'Valves'. The 'Fan' option is selected. The main panel shows various configuration settings for the fan control.

Relay Management Type: Switching (only 1 relay on for each speed)

Delay between Fan Speed Switching: 3 ds

Fan Control Mode: Automatic

Minimum Fan Speed when the Valve is Open: 1

AUTOMATIC FAN CONTROL

Input for Automatic Fan Control: PI Control (Continuous)

Threshold Values to Change the Fan Speed

(Speed 1 if Control < Threshold 2)

Threshold 2: 50 %
(Speed 2 if Control >= Threshold 2)

Threshold 3: 75 %
(Speed 3 if Control >= Threshold 3)

Note: In Automatic Mode the Fan Speed will be set to 0 when the Valve is Closed

Delays for Fan Activation/Deactivation: ☐

Starting Characteristic of Fan: ☐

Status Objects: ☐

Figure 2. Valve-oriented control: Fan.

2.2.1.1 AUTOMATIC CONTROL

The automatic fan speed control may be subject to an **external PI control** (performed by a thermostat) or to the **difference between a reference temperature and a setpoint**, both received externally as well.

● **PI Control:**

The PI value is received through a percentage communication object, and may be the same value that controls the valve in case it has been configured to be controlled through a PI control value too.

The application programme will apply one speed or another to the fan depending on whether such PI value exceeds certain parameterisable thresholds, referred to as **Threshold 1**, **Threshold 2** and **Threshold 3**.

- PI Value < Threshold 1 → the fan turns off.
- PI Value ≥ Threshold 1 → speed 1.
- PI Value ≥ Threshold 2 → speed 2
- PI Value ≥ Threshold 3 → speed 3

The above criterion, as well as the number of thresholds to be parameterised, may be conditioned to the value assigned to the parameter **Minimum Fan Speed When the Valve is Open** (section 2.2.1).

● Temperature Difference:

The fan speed is determined according to the difference between a **temperature setpoint** (or target temperature) and an **ambient temperature** (or reference temperature), both received through their specific objects.

Such difference determines the target speed depending on whether it exceeds or not certain parameterisable thresholds, referred to as **Dif 1**, **Dif 2** and **Dif 3**:

- $|T_{\text{setpoint}} - T_{\text{ambient}}| < \text{Dif 1} \rightarrow \text{the fan turns off}$.
- $|T_{\text{setpoint}} - T_{\text{ambient}}| \geq \text{Dif 1} \rightarrow \text{speed 1}$.
- $|T_{\text{setpoint}} - T_{\text{ambient}}| \geq \text{Dif 2} \rightarrow \text{speed 2}$.
- $|T_{\text{setpoint}} - T_{\text{ambient}}| \geq \text{Dif 3} \rightarrow \text{speed 3}$.

The above criterion, as well as the number of thresholds to be parameterised, may be conditioned to the value assigned to the parameter **Minimum Fan Speed When the Valve is Open** (section 2.2.1).

An additional parameter to set a **hysteresis** over the above thresholds is also provided. This helps to avoid continuous fan speed switches in case the ambient temperature keeps fluctuating around the limit temperature between contiguous levels.

The following example illustrates this control type.

Example (temperatures difference)

Suppose a hysteresis of 0.5°C and the following thresholds:

- Dif 1 = 1°C .
- Dif 2 = 3°C .
- Dif 3 = 5°C .

The current mode is “Cooling” and the temperature setpoint is 25°C . Given the following ambient temperature values, the fan speed levels will be triggered as indicated:

- **26.4°C** → $|T_{\text{setpoint}} - T_{\text{ambient}}| < (\text{Dif } 1 + h) \rightarrow$ The fan turns off.
- **27°C** → $|T_{\text{setpoint}} - T_{\text{ambient}}| > (\text{Dif } 1 + h) \rightarrow$ Speed 1.
- **30.5°C** → $|T_{\text{setpoint}} - T_{\text{ambient}}| \geq (\text{Dif } 3 + h) \rightarrow$ Speed 3.
- **28.6°C** → $|T_{\text{setpoint}} - T_{\text{ambient}}| > (\text{Dif } 2 - h) \rightarrow$ Speed 2.

ETS PARAMETERISATION

On the first hand, it is necessary to configure the following parameter:

- **Input for Automatic Fan Control:** “PI Control (Continuous)” or “Temperature Difference”.

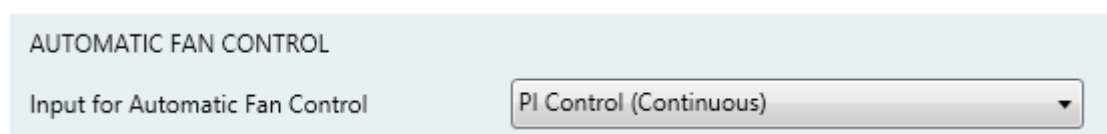


Figure 3. Automatic Fan Control.

In case of selecting “PI Control (continuous)”, the integrator should enter the desired values (in terms of percentage) for **Threshold 1**, **Threshold 2** and **Threshold 3**, or those that may be required depending on the minimum fan speed configured.

Threshold Values to Change the Fan Speed	
(Speed 0 if Control < Threshold 1)	
Threshold 1 (Speed 1 if Control >= Threshold 1)	25 %
Threshold 2 (Speed 2 if Control >= Threshold 2)	50 %
Threshold 3 (Speed 3 if Control >= Threshold 3)	75 %

Figure 4. Automatic Fan Control. PI Control (Continuous).

In case of selecting “Temperature Difference”, the integrator should enter the desired values (in terms of tenths of a degree) for **Diff 1**, **Diff 2** and **Diff 3**, or those that may be required depending on the minimum fan speed configured. A **hysteresis** (between 0 and 20 tenths of a degree) can also be parameterised, if required.

Difference between Ambient and Setpoint Temperature Thresholds	
(Speed 0 if Difference < Diff. 1)	
Diff 1 (Speed 1 if Difference >= Diff 1)	20 x0.1 °C
Diff 2 (Speed 2 if Difference >= Diff 2)	40 x0.1 °C
Diff 3 (Speed 3 if Difference >= Diff 3)	60 x0.1 °C
Hysteresis	0 x0.1 °C

Figure 5. Automatic Fan Control. Temperature Difference.

To perform the temperature difference, two 2-byte objects are provided: “[FCn] **Ambient Temperature**” and “[FCn] **Setpoint Temperature**”.

2.2.1.2 MANUAL CONTROL

The manual fan control can be performed **through different kinds of objects** that allow the user select the desired fan speed without taking into account neither the time that the valve remains open nor the temperature differences. However, the following remarks do apply to the manual fan control.

- If the valve is open and a **minimum fan speed** other than zero has been configured, the user will not be able to set a speed below it.

- If the fan coil is in Heating mode, the user will not be able to turn on the fan while **the valve remains closed**.

As the manual fan control and the objects provided to perform it are common for both, valve-oriented control and fan-oriented control, the additional details have been included into section 2.4.

2.2.1.3 AUTOMATIC + MANUAL CONTROL

When both the manual fan control and the automatic fan control are permitted, everything explained in sections 2.2.1.1 and 0 will still applicable, however, some additional options will be available, as detailed next.

PARAMETRIZACIÓN ETS

The following specific parameters show up after selecting “Automatic + Manual” for the fan control mode:

- **Fan Control After ETS Download:** “Automatic” or “Manual”.
- **Automatic/Manual Switch Object:** sets the particular values that will switch from one control mode to the other, if received through object “[FCn] Fan: Manual / Automatic”:
 - “0 = Automatic; 1 = Manual”
 - “0 = Manual; 1 = Automatic”.
- **Return to Automatic Mode after a Period of Time:** sets if the manual control switches automatically to automatic control after a certain time of inactivity, configured in “**Duration of Manual Override of Fan**” (1 to 1440 minutes; 1 to 24 hours). It may be overwritten through the homonymous object.

Fan Control Mode	Automatic + Manual
Fan Control after ETS Download	Automatic
Automatic/Manual Switch Object	0 = Automatic; 1 = Manual
Return to Automatic Mode after a Period of Time	☒
Duration of Manual Override of Fan	30
	min
Minimum Fan Speed when the Valve is Open	1

Figure 6. Automatic + Manual Control.

2.2.2 VALVE SETTINGS

The valves (or the valve, if there are only two pipes) may be controlled by two alternative approaches, depending on the type of the communication object (one-byte or binary) used by the external thermostatic controller to send the orders:

- **PWM Control** (Pulse Width Modulation): 1 bit.

The control variable sends **binary values**, which determine whether the valve should remain open or closed.

- **PI Control** (Proportional-Integral): 1 byte.

The control variable sends **percentage values**, which determine the portion of time the valve should remain open every cycle. For instance, a value of 50% means the valve must remain open for half the cycle time.

Being the valves on/off type, the second case implies **controlling them through PWM signals** as well, although calculated according to the percentage value. It is therefore necessary to parameterise the specific cycle time desired for that PWM control. In the first case, on the contrary, such cycle time is not necessary with the exception of the **scene management** (section 2.7; **Error! No se encuentra el origen de la referencia.**).

ETS PARAMETERISATION

When the control type has been configured as applied to the valve, the “Valve” tab (or “Valves”, if the fan coil consists of four pipes) shows the following specific options (for any other parameters, please refer to section 2.5):

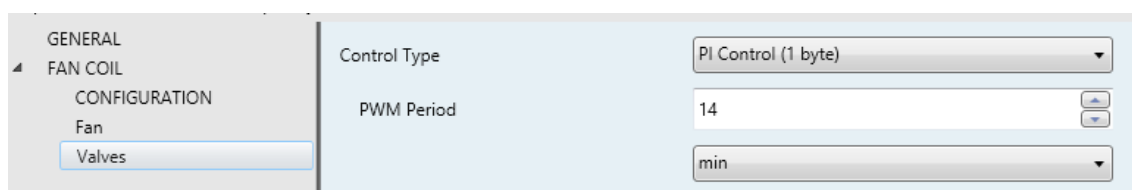


Figura 7. Control type options for the valve.

● **Control Type:** “PI Control (1 byte)” or “Control PWM (1 bit)”.

- PI Control (1 byte): enables the objects “[FCn] Cooling Valve: PI Control (Continuous)” and / or “[FCn] Heating Valve: PI Control (Continuous)”, set aside for the reception of PI control percentage values from the KNX bus.
- PWM Control (1 bit): enables the objects “[FCn] Cooling Valve: PWM Control (1 bit)” and / or “[FCn] Heating Valve: PWM Control (1 bit)”, intended for the reception of open / close orders for the valve from the KNX bus.

In any of the two cases, the objects “[FCn] Cooling Valve (Status)” and “[FCn] Heating Valve (Status)” (or, in case of only two pipes, a single object named “[FCn] Valve (Status)”) will be available. Their value will be “1” while the valve is open, and “0” while it remains closed.

- **PWM Period:** sets the cycle time for the PWM control (3 – 1440 minutes, or 1 – 24 hours). In case of having selected “PWM Control (1 bit)” in the above parameter, the period is inherent to the control signal received, and therefore this parameter is only required for scene management.

2.3 FAN-ORIENTED CONTROL

In this case, the control will focus on responding to the orders over the fan that arrive from the bus, making the valve state depend on these orders.

Certain settings must be performed with independence of the control type selected. It is therefore advisable to read sections 2.4 and 2.5 as well.

2.3.1 FAN SETTINGS

The fan control may be **manual** (the fan speed will be controlled externally), **automatic** (the fan speed will be controlled by the module itself), or **both**, although the automatic control mode will be available at any time (the manual control mode needs to be enabled by parameter).

ETS PARAMETERISATION

The screenshot shows the ETS parameterisation interface for Fan-oriented control. On the left, a sidebar contains the following menu items: GENERAL, FAN COIL, CONFIGURATION, Fan (highlighted), and Valves. The main area is titled 'Fan' and contains the following settings:

- Relay Management Type:** A dropdown menu set to 'Switching (only 1 relay on for each speed)'.
- Delay between Fan Speed Switching:** A numeric input field set to '3' with a unit dropdown set to 'ds'.
- Enable Manual Fan Control:** A checkbox that is currently unchecked.
- Threshold Values to Change the Fan Speed:** A section with three thresholds:
 - Threshold 1:** (Speed 0 if Control < Threshold 1) set to '25' %.
 - Threshold 2:** (Speed 2 if Control >= Threshold 2) set to '50' %.
 - Threshold 3:** (Speed 3 if Control >= Threshold 3) set to '75' %.
- Delays for Fan Activation/Deactivation:** A checkbox that is currently unchecked.
- Starting Characteristic of Fan:** A checkbox that is currently unchecked.
- Status Objects:** A checkbox that is currently unchecked.

Figure 8. Fan-oriented control: fan.

When the control type has been configured as applied to the fan, the “Fan” tab shows the following specific option (for the remaining parameters, please refer to section 2.4).

- **Enable Manual Fan Control:** sets whether the manual fan control will be required or not (see section 2.3.1.2).

On the other hand, the automatic fan control is available at any time.

2.3.1.1 AUTOMATIC CONTROL

The automatic fan speed control will be subject to the value of one specific **percentage** object per working mode (heating / cooling).

It is therefore required to set the **range of the percentage values** that will determine the fan speeds established when received from the bus (from an external, continuous PI thermostat controller). For this purpose, the value of Threshold 1, Threshold 2 and Threshold 3 is defined in parameters, so that:

- Value received < Threshold 1 → the fan is turned off.
- Value received $\geq$ Threshold 1 → speed 1.
- Value received $\geq$ Threshold 2 → speed 2
- Value received $\geq$ Threshold 3 → speed 3

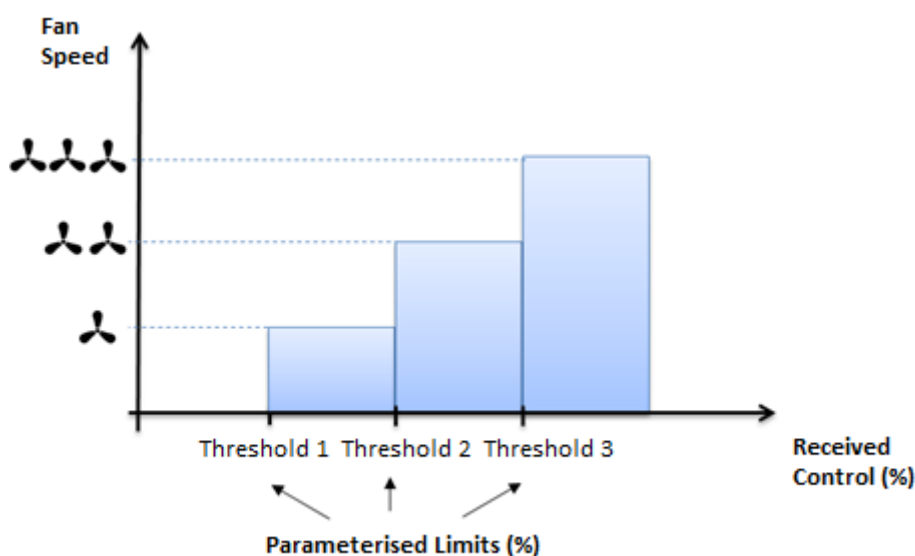


Figure 9. Parameterised thresholds, received control and fan speed relationship.

The above criterion, as well as the number of thresholds to be parameterised, may be conditioned to the value assigned to the parameter “**Number of fan speeds**” (section 2.1).

ETS PARAMETERISATION

The value of **Threshold 1**, **Threshold 2** and **Threshold 3** are established through the following parameters:

The screenshot shows a configuration window titled "Threshold Values to Change the Fan Speed". It contains three rows, each for a different threshold. Each row has a text label, a numerical input field, and a percentage sign. The input fields are set to 25, 50, and 75 respectively. To the right of each input field are two small square buttons with up and down arrows.

Threshold	Value	Unit
Threshold 1 (Speed 0 if Control < Threshold 1)	25	%
Threshold 2 (Speed 1 if Control >= Threshold 2)	50	%
Threshold 3 (Speed 3 if Control >= Threshold 3)	75	%

Figure 10. Threshold values to change the fan speed.

The automatic control is performed through the percentage objects “[FCn] Heating Fan: Continuous Control” and/or “[FCn] Cooling Fan: Continuous Control”, which are typically intended to be linked to an external PI thermostatic control.

2.3.1.2 AUTOMATIC + MANUAL CONTROL

The (optional) manual control mode lets the user set a fan speed **from their own user interface** with independence of the value received through the objects of the automatic fan control, which are typically linked to an external thermostat.

An object is provided to **switch from the automatic control mode to the manual one**, as well as an option to switch back to the automatic control when any manual orders have been received after a certain time.

It is also possible to select the desired reaction when, being active the manual control mode, **a percentage value lower than Threshold 1 is received through the automatic control object** (see section 2.3.1.1). Under the automatic control mode this would imply closing the valve and turning off the fan, but under the manual control the available options are:

- **Close the valve and turn off the fan**, to ensure a minimum thermostat control.
- **Ignore the order**, thus giving always priority to the manual orders even if the temperature setpoint has already been reached. Therefore, the possibility of having the valve closed and the fan off is not avoided.

- **Close the valve and turn off the fan, but only under Heating**, thus maintaining the user control but preventing that the fan coil unit is left with the heating valve closed and the fan on (which may make the user feel coolness).

In the latter two cases, once the fan is turned off and the valve is closed:

- While this turned-off state lasts, any manual orders will be ignored and responded with the current speed (0) and control (automatic) statuses.
- After a percentage value higher than Threshold 1 is received, the corresponding speed will be applied, and the thermostatic control resumed. However, if a new manual order is received afterwards, the manual control will be activated again.

The remaining options are explained in section 2.4 as they are common to both, a valve-oriented control and a fan-oriented control.

ETS PARAMETERISATION

The following specific parameters show up after enabling the manual control mode.

Figura 11. Manual fan control for applied to the fan control configuration.

- **Automatic/Manual Switch Object:** sets the values that should be received through “[FCn] Fan: Manual/Automatic” to switch the control mode: “0 = Automatic; 1 = Manual” or “0 = Manual; 1 = Automatic”.
- **Behaviour for a Continuous Control Order PI < Threshold 1:** sets whether during the manual control mode, automatic switch-off orders should be handled: “Close Valve and Turn Off Fan”, “Nothing (Ignore PI Order in Manual Control)” or “Close Valve and Turn Off Fan (Only in Heating)”.

- **Return to Automatic Mode after a Period of Time:** sets if the manual control should automatically switch to automatic control after a certain time of inactivity, configured in “**Duration of Manual Override of Fan**” (1 to 1440 minutes; 1 to 24 hours). It may be overwritten through the homonymous object.

The remaining parameters in this tab have already been described for the automatic control mode (see section 2.3.1.1) or are common for both, the valve-oriented control and the fan-oriented control and therefore are detailed in section 2.4.

2.3.2 VALVE SETTINGS

The valve state in this case will be **determined by the state of the fan**, both under an automatic fan control (through the PI object) and under a manual fan control (through any of the objects provided for that purpose).

- **Automatic fan control (PI):**

The valve of the current mode (heating / cooling) **will remain open** as long as the fan is turned on, or otherwise closed.

- **Manual fan control:**

The behaviour will be similar, however, even being the manual fan control active, a PI order lower than Threshold 1 may not be ignored and thus make the valve close and the fan turn off (if set in parameters; see section 2.3.1.2), although the manual control will still remain active. Therefore, a later manual order superior to switch to a speed other than 0 will imply opening back the valve and applying the requested speed.

ETS PARAMETERISATION

When a fan-oriented control has been configured, the “Valve” tab does not show any specific parameters, as all of them are common for a valve-oriented control. See section 2.5.

In addition, any object to directly manipulate the valves is provided in this case, as their state is subject to the state of the fan and to the object that sets the current working mode (heating / cooling).

2.4 COMMON SETTINGS (FAN)

This section details the fan settings that are common for both control types: valve-oriented and fan-oriented.

The 'Relays' Fan Coil module allows controlling up to **three fan speed levels**. To activate each of them, one binary output becomes active, either through *switching* or through *accumulation*:

- **Switching**: only one of the fan control binary outputs is active each time, i.e., only one relay is closed for the activation of each speed. It is possible to set up a **delay** between the opening of the source speed relay and the closing of the target speed relay (so both stay open for some instants).
- **Accumulation**: the speed will be proportional to the number of active outputs (i.e. closed relays), which therefore get triggered in sequence.

Moreover, the previous sections have already described the possibility of performing an automatic fan control and a manual fan control. Although some differences apply depending on whether the fan coil control is valve-oriented or fan-oriented (see previous sections), the most significant difference is that:

- Under an **automatic control**, the fan speed is calculated automatically according to the parameterisation and to an external reference.
- Under a **manual control**, the user can directly take part in the selection of the desired speed through a set of communication objects of different types.

When both control modes are allowed, an object is provided to switch from one mode to the other, although **the reception of a manual order** causes itself a switch to the manual control mode. Besides, it is possible to configure a time-out after which, if no additional manual orders have been received, the control will switch back to automatic.

Regarding the **manual control**, the communication objects that permit commuting the fan speed are of the following types, and are in any case conditioned by the number of fan speeds allowed (see section 2.1):

- **One-bit objects (one per speed)**, which activate a particular speed level on the reception of the value “1”.
- **Step-control objects**: one-bit objects for increasing or decreasing the speed level sequentially, either **cyclically** (a further step once reaching the maximum level activates the minimum level again) or not.

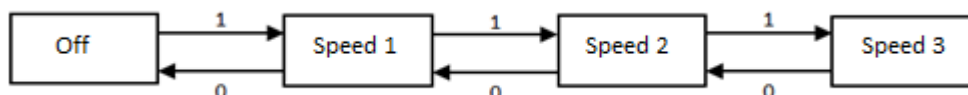


Figure 12. Non-cyclical fan step control.

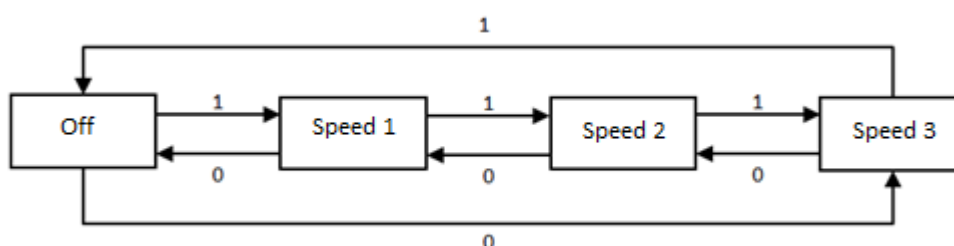


Figure 13. Cyclical fan step control.

The above sequence can incorporate an additional state: **the automatic speed mode**, that allows the automatic pass to automatic control. The following options are possible:

➤ Non-cyclical control:

- Activate automatic speed mode if, being the fan off, a request to increase the speed is received.
- Activate automatic speed mode if, being the fan at the maximum speed level, a request to increase the speed level is received.
- Activate the automatic speed mode in any of the above two.

➤ Cyclical control: the automatic speed will be an intermediate state between the two ends of the sequence.

- **One-byte enumerated object**: the speed switch will take place upon the arrival of the proper integer value (0, 1, 2, 3).
- **Percentage object**: the speed switch will take place upon the arrival of the proper percentage value; according to the KNX standard (see Table 2).

Available speeds	Speed	Percentage
One speed	0	0%
	1	1% – 100%
Two speeds	0	0%
	1	0.4% – 50.2%
	2	50.4% – 100%
Three speeds	0	0%
	1	0.4% – 33.3%
	2	33.7% – 66.6%
	3	67% - 100%

Table 2. Manual control through the percentage object

For all the aforementioned objects (with the exception of the step-control object) **an analogous status object** is available, making it possible to request the current fan speed at any time, even during the automatic control.

On the other hand, it is possible to **delay the activation and deactivation of the fan** after an action over the valve (for either the Heating or the Cooling modes):

- Under a **valve-oriented control**, this delay will be applied to the automatic switch-on / switch-off of the fan when the valve is opened or closed.
- Under a **fan-oriented control**, this delay will be applied to the switch-on of the fan when such order is received being the fan off and the valve closed (this will imply that the valve opens), as well as to the switch-off of the fan when such order is received being the fan on and the valve open fan (this will imply that the valve closes).

Example: *delay in the activation / deactivation.*

Given a valve-oriented control with a delay of two minutes for the activation of the fan and of ten minutes for the deactivation, and assuming an automatic fan control, a PWM value of 50% is received.

- *On the arrival of the order, the valve will open immediately.*
- *Two minutes later, the fan turns on at the speed that may correspond to the parameterisation.*
- *After half of the PWM cycle, the valve closes.*
- *Ten minutes later, the fan turns off.*

Finally, either under a valve-oriented control or a fan-oriented control, a **fan engine starting characteristic** may be configured, which is useful for some engines that require an extra amount of current in comparison to that required in normal operation. Thus, during the start-up, some engines need to step through a higher speed level (e.g. 2 or 3) for some time before they switch to lower speeds

ETS PARAMETERISATION

The following parameters can be found in the “Fan” tab either under a valve-oriented control and under a fan-oriented control (please refer to sections 2.2 and 2.32.3 for the specific parameters of each case).

- **Relay Management Type:** “Switching (only 1 relay on for each speed)” or “Accumulation (multiple relays on)”.

If set to “Switching”, an additional parameter (“**Delay between Fan Speed Switching**”) is shown to let the integrator configure a delay (3 to 100 tenths of a second) since the source valve opens and the target valve closes, thus making both valves remain open during the delay configured.

- **Delays for Fan Activation / Deactivation:** enables / disables an additional parameter tab named “**Delays**”, which contains the following parameters:

COOLING

Delay for Fan Activation when Valve Opens (0 = No Delay) 0 s

Delay for Fan Deactivation when Valve Closes (0 = No Delay) 0 s

HEATING

Delay for Fan Activation when Valve Opens (0 = No Delay) 0 s

Delay for Fan Deactivation when Valve Closes (0 = No Delay) 0 s

Figure 14. Parameters in “Delays” tab.

- **Delay for Fan Activation when Valve Opens:** sets the amount of time (0 to 3600 seconds, or 0 to 1440 minutes, or 0 to 24 hours) the fan will delay its switch-on after the valve is opened.
- **Delay for Fan Deactivation when Valve Closes:** sets the amount of time (0 to 3600 seconds, or 0 to 1440 minutes, or 0 to 24 hours) the fan will delay its switch-off after the valve is closed.

The above two parameters should be configured for **Heating**, for **Cooling** or for both, depending on the configuration made so far.

- **Starting Characteristic of Fan:** enables / disables an additional parameter tab named “**Starting Characteristic**”, which contains the following parameters:

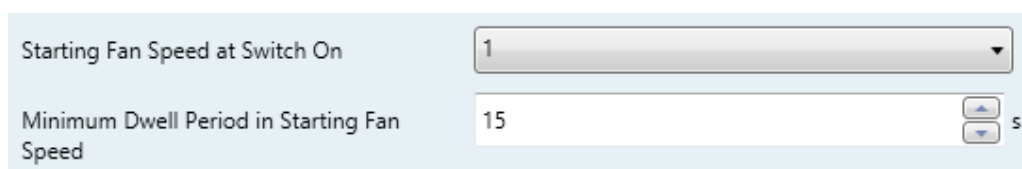


Figure 15. Parameters in “Starting Characteristic” tab.

- **Starting Fan Speed at Switch On:** sets the fan speed level (“1”, “2” or “3”) the fan engine should adopt when it starts up, before being able to adopt lower speeds. Note that the available options depend on the number of fan speed levels enabled in the “Configuration” tab (see section 2.1).
- **Minimum Dwell Period in Starting Fan Speed:** sets the time (1 to 250 seconds) the fan engine should remain at the above speed when it starts up.

- **Status Objects:** enables / disables an additional parameter tab named “**Status Objects**”, which permits enabling different fan status objects:

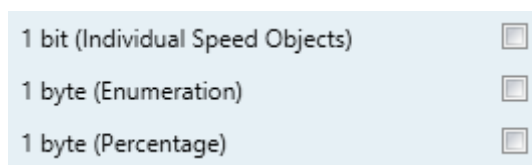


Figure 16. Parameters in the “Status Objects” tab.

- **1 bit (Individual Speed Objects):** enables/disables the “[FCn] Fan: Speed x (Status)” binary objects, where “x” runs from 0 to 3. These

objects throw the value “1” when the speed level they refer to is active, and “0” when not.

- **1 byte (Enumeration):** enables/disables the “[FCn] Fan: Enumeration Speed (Status)” one-byte object, which will throw values from 0 to 3 according to the current fan speed.
- **1 byte (Percentage):** enables/disables the “[FCn] Fan: Percentage Speed (Status)” one-byte object, which will throw values from 0% to 100% according to the current fan speed and according to the KNX standard (see Table 2).

2.5 COMMON SETTINGS (VALVE)

This section details the valve settings that are common for both control types: valve-oriented and fan-oriented.

On the first hand, it is necessary to set the **value that defines the open state** of the valve, i.e., whether it is the value “0” or the value “1” the one to be interpreted as “open valve” in the objects related to the valve (PWM control, status, etc.).

Equally, it is possible to parameterise a **minimum time lapse between successive valve switches**, which may help to prevent damage due to quick and successive commutations of the valve. It is important to configure this time with caution, as it may improperly delay the operation of the valve, for example if configuring, under a valve-oriented control driven by a continuous PI signal, a cycle time (see section 2.2.2) lower than this minimum time.

It is also possible to set up a **delay over the mode switches** (Cooling / Heating), provided that both modes are available (see section 2.1). This allows delaying (always) the actual opening of the valve of the target mode since the reception of the mode switch order, so that:

- In a **four-pipe** fan coil, it is possible to assure that the two valves do not remain open at the same time when closely-spaced mode switch orders are received.
- In a **two-pipe** fan coil, it is possible to assure that the valve does remain closed before opening again when closely-spaced mode switch orders are received. In this case, the actual delay may be increased if a minimum time lapse between successive valve switches has been configured.

Finally, the configuration of the valve allows **enabling a purge function** (one per valve), which consists in opening the selected valve, closing the opposite valve (if open) and stopping the fan for a configurable time space.

The purge **begins** when one “1” is received through the corresponding object, and **ends** when the parameterised time elapses, as well as when one “0” arrives, a bus failure takes place, the fan coil control module is turned off, or the user enters the Test On mode of the device that contains the fan coil control module. To restart the process (from the beginning), the corresponding object should be written again with a “1”.

Notes:

- Any control order and any mode switch request received during the purge function **will be executed after the purge ends**. Thus, as soon as the orders are received, they will be responded with the current status, making it evident that they will not be performed for the moment.
- If no control orders are received, once the purge is over the fan coil control module **will recover previous state**.
- It will not be possible to close the opposite valve if both valves are under the purge process. **Purge concurrence** is allowed, but this implication should be beard in mind.

ETS PARAMETERISATION

The following parameters can be found in the “Valve” tab (or “Valves”, if the fan coil consists of four pipes) either under a valve-oriented control and under a fan-oriented control (please refer to sections 2.2 and 2.3 for the specific parameters of each case).

- **Value for Open Valve:** “1” or “0”.
- **Minimum Time Between Valve Switches:** either 0 to 3600 seconds, or 0 to 1440 minutes, or 0 to 24 hours.
- **Minimum Delay for Mode Change** (only if Cooling as well as Heating are available): same range as the above parameter.
- **Enable Valve Purge:** enables / disables the valve purge function, as well as the object “[FCn] Valve: Trigger Purge” (which must be written with the value “1” to start the purge, and with the value “0” to afterwards interrupt it) and the following additional parameters:
 - **Duration of the Valve Purge:** time lapse the valve will remain in the purge state once the function has been triggered (either 0 to 3600 seconds, or 0 to 1440 minutes, or 0 to 24 hours).
 - **Enable Status Object:** adds the object “[FCn] Valve: purge (status)” to the project topology, which permits reading the object to obtain whether the purge is currently in progress (value “1”) or not (value “0”).

Enable Valve Purge	☒
Duration of Valve Purge	10
(1 - 1440)	
	min
Enable Status Object	☒

Figure 17. Configuration of the valve purge.

2.6 CYCLICAL MONITORING

It is possible to perform a **cyclical monitoring** of the control orders that are received from the external device (e.g.: a thermostat) that controls the fan coil module, with the aim of detecting communication issues.

It is necessary to define a **cyclical monitoring period**, in other words, how frequently these orders are expected to arrive from the bus. If this period is exceeded with no orders received, a certain reaction will be performed and the error will be notified through a specific object.

The **reaction in case of an error** can consist in:

- Closing the valve and turning off the fan,
- Executing a scene (as if received from the bus),
- Doing nothing.

The **notification object** will keep sending (every 15 minutes) the value “1” while the error persists, and will send the value “0” (only once) once solved, that is, once the external orders are resumed or when the module enters a state that deactivates the cyclical monitoring:

- The cyclical monitoring only takes place while the module has been turned on and the valves are not being purged (see section 2.5).
- Under a fan-oriented control, the monitoring will take place unless the **manual control** mode of the fan is active.

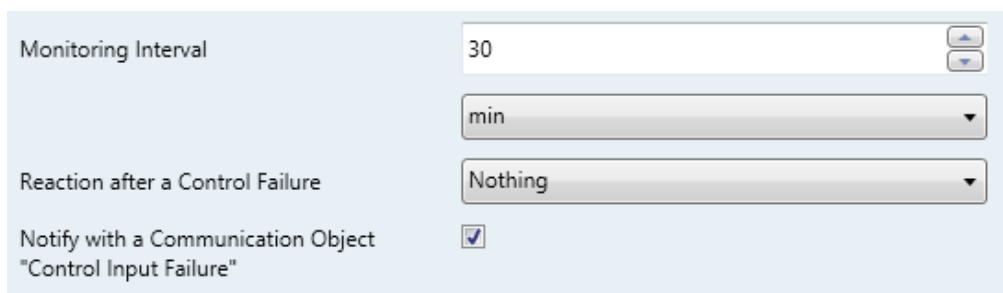
Note: *under a valve-oriented control, the monitoring period must be at equal or greater than the PWM control period.*

ETS PARAMETERISATION

After enabling the cyclical monitoring function from the “Configuration” tab (section 2.1), the “Cyclical Monitoring” tab will be available, containing the following parameters.

- **Monitoring Interval:** either 10 to 3600 seconds, or 1 to 1440 minutes, or 1 a 24 hours.

- **Reaction after a Control Failure:** “Nothing”, “Close Valve and Turn Off Fan” or “Execute Scene”. If the latter is selected, an additional parameter appears:
 - **Scene:** sets which of the up to four scenes defined (see section 2.7) will be executed.
- **Notify with a Communication Object:** enables the “[FCn] Control Input Failure” binary object, which will send the value “1” every 15 minutes in case of a communication error, and the value “0” (only once) once the error is over.



Monitoring Interval	30
	min
Reaction after a Control Failure	Nothing
Notify with a Communication Object "Control Input Failure"	☒

Figura 18. Cyclical Monitoring.

2.7 SCENES

It is possible to define up to **four scenes** so that, when the corresponding value is received from the bus, the module adopts a certain state, which must be defined according to the following

- On / Off state of the module.
- Under a valve-oriented control:
 - Value of the valve PI control (in percentage)
 - Automatic fan control mode.
 - Manual fan control mode.
 - Fan speed.
- Under a fan-oriented control:
 - Fan speed.
 - Valve state.

It is important to bear in mind that executing a scene is equivalent to sending the analogous orders to the corresponding objects. Therefore, the result will depend on the initial state of the fan coil module when the scene is executed.

For example, if a manual speed selection order is received during the purge process of the valve, the order will be buffered and executed after the completion of the purge, as it would happen in case of receiving the request through the analogous communication object.

On the other hand, if the scene in execution implies turning the fan coil module off, the response will be immediate, and the purge process will be aborted.

This module does not permit scene recording.

ETS PARAMETERISATION

Once Scenes has been enabled from the “Configuration” tab (see section 2.1), a new tab named “Scenes” is added to the tab tree on the left, containing the following options:

Scene 1	☒
Scene Number	1
Fan Coil State	On
Valve PI Control	0
Fan Control	Manual
Fan Speed	1

Figure 19. Scenes (valve-oriented control).

- **Scene “n”**: enables or disables scene “n”, which should be then configured through the following additional parameters:

- **Scene number**: sets the value (1-64) that, when received through object “[Fan coil] Scenes”, will trigger the adoption of the configuration defined next.
- **Fan Coil State**: may be set to “On” or “Off”.

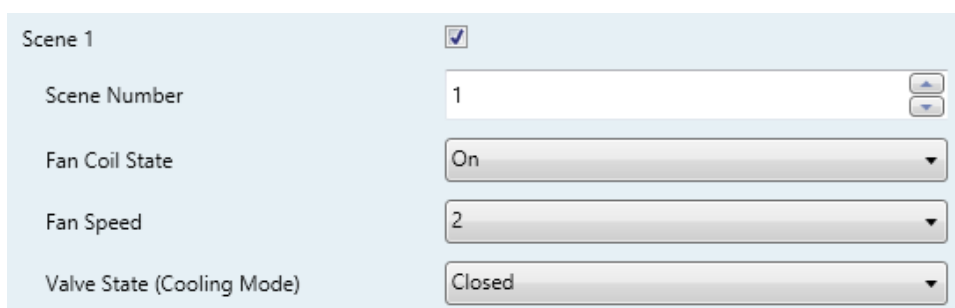
The following parameters only apply under a **valve-oriented** control, and provided that the Fan Coil State has been set to “On”:

- **Valve PI Control**: sets (in percentage) the value of the PI control to be adopted.
- **Fan Control** (only available if both, the automatic fan control and the manual fan control have been enabled; see section 2.2.1): may be set to “Automatic” or “Manual”.
- **Fan Speed** (only available if the above parameter has been set to “Manual”, or if only the manual fan control is available): may be set to “0”, “1”, “2” o “3”, assuming that all of them are available.

The parameters only apply under a **fan-oriented** control, and provided that the Fan Coil State has been set to "On":

- **Fan Speed:** "0", "1", "2" or "3", assuming that all of them are available.
- **Valve State** (only available if the above speed has been set to a value other than "0"): "Close" or "Open".

Note: *this option only applies to the **Cooling** mode. In case of being in the Heating mode during the execution of the scene, the state of the valve will be subject to the state of the fan (closed valve if the fan is off, and open valve if the fan is on).*



Scene 1	☒
Scene Number	1
Fan Coil State	On
Fan Speed	2
Valve State (Cooling Mode)	Closed

Figure 20. Scenes (fan-oriented control).

2.8 INITIALISATION

Regarding the initialisation of the fan coil module after a download from ETS or a bus failure, it is possible to choose between a custom configuration and the default one.

For a custom initialisation, it is necessary to define the following:

- **The initial state**, which may be off, or the previous one (after a download, it will be always off).
- **Whether to send the status objects to the bus**, so other devices in the installation get updated.

Any subset of the following objects can be sent (supposing that the functionality they refer to has been enabled in the configuration):

- On/Off.
- Mode (Heating / Cooling).
- Fan Speed (all enabled objects will be sent).
- Fan Control Mode (Automatic / Manual).
- State of the Valves (Closed / Open).
- Purge State of the Valves (Active / Not Active).
- Control Input Failure.

It is also possible to define a **delay** for this sending.

In the default initialisation no objects are sent, and the module will recover the previous state (off, after a download).

ETS PARAMETERISATION

If the option to customise the initialisation has been selected in the “Configuration” tab (section 2.1), the tab tree will include a new tab named “Initialization”, which will contain the following parameters:

- **Initial State**: “Previous” or “Off”.

- **Send Status Objects:** if activated, several checkboxes will show up to select the specific objects to be sent, as well as the following parameter:
 - **Delay:** sets the delay, since the start-up of the device, after which the objects will be sent (0 to 600 seconds).

Initial Status	Previous
Send Status Objects	☒
On/Off	☐
Mode (heating/cooling)	☐
Fan Speed	☐
Fan Mode (automatic/manual)	☐
Valve (closed/opened)	☐
Valve Purge (active/no active)	☐
Control Input Failure	☐
Delay (0 = No Delay)	0 s

Figura 21. Initialisation.

Únete y envíanos tus consultas
sobre los dispositivos Zennio:
<http://zennio.zendesk.com>

Zennio Avance y Tecnología S.L.
C/ Río Jarama, 132. Nave P-8.11
45007 Toledo (Spain).

Tel. +34 925 232 002.

Fax. +34 925 337 310.

www.zennio.com

info@zennio.com



RoHS